Computing Approximate (Symmetric Block) Rational Krylov Subspaces without Explicit Inversion

Thomas Mach   *Miroslav S. Pranić*
Raf Vandebril

Report TW 636, September 2013

KU Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

Computing Approximate (Symmetric Block) Rational Krylov Subspaces without Explicit Inversion

Thomas Mach   *Miroslav S. Pranić*
Raf Vandebril

Report TW 636, September 2013

Department of Computer Science, KU Leuven

Abstract

It has been shown that approximate extended Krylov subspaces can be computed –under certain assumptions– without any explicit inversion or system solves. Instead the necessary products $A^{-1}v$ are obtained in an implicit way retrieved from an enlarged Krylov subspace. In this paper this approach is generalized to rational Krylov subspaces, which contain besides poles at infinite and zero also finite non-zero poles.

Also an adaption of the algorithm to the block and the symmetric case is presented. For all variants of the algorithm numerical experiments underpin the power of the new approach. Rational Krylov subspaces can be used, e.g., to approximate matrix functions or the solutions of matrix equations.

Keywords : Krylov, rational Krylov, iterative methods, Ritz-values, rational polynomial approximation, rotations, QR factorization
MSC : Primary : 65F60, Secondary : 65F10, 47J25, 15A16.

COMPUTING APPROXIMATE (SYMMETRIC BLOCK) RATIONAL KRYLOV SUBSPACES WITHOUT EXPLICIT INVERSION *

THOMAS MACH†, MIROSLAV S. PRANIĆ‡, AND RAF VANDEBRIL†

Abstract. It has been shown that approximate extended Krylov subspaces can be computed –under certain assumptions– without any explicit inversion or system solves. Instead the necessary products $A^{-1}v$ are obtained in an implicit way retrieved from an enlarged Krylov subspace. In this paper this approach is generalized to rational Krylov subspaces, which contain besides poles at infinite and zero also finite non-zero poles.

Also an adaption of the algorithm to the block and the symmetric case is presented. For all variants of the algorithm numerical experiments underpin the power of the new approach. Rational Krylov subspaces can be used, e.g., to approximate matrix functions or the solutions of matrix equations.

Key words. Krylov, rational Krylov, iterative methods, Ritz-values, rational polynomial approximation, rotations, QR factorization

AMS subject classifications. 65F60, 65F10, 47J25, 15A16

1. Introduction.

In [17] we have presented a way of computing approximate extended Krylov subspaces. This approach computes the vectors $A^{-k}v$ in an implicit way without any explicit computation of A^{-1} or any explicit system solve. We have shown that for many examples the approximation is of good quality. Here we generalize this algorithm to rational block Krylov subspaces. Further we will explain how to use and preserve symmetry.

Let $A \in \mathbb{C}^{n\times n}$ be a matrix and $v \in \mathbb{C}^n$ be a vector, the space spanned by

$$\mathcal{K}_m(A, v) = \operatorname{span}\left\{v, Av, A^2v, \ldots, A^{m-1}v\right\}.$$

is called a *Krylov subspace*. Krylov subspaces have been used in a wide range of applications, e.g., for the solution of sparse indefinite systems [20], for large unsymmetric systems [25], or Lyapunov equations [11]. Rational Krylov subspaces have been introduced by Ruhe in [21] and were later investigated in [22–24]. Let $\sigma = [\sigma_1, \sigma_2, \ldots, \sigma_{m-1}]$, with $\sigma_i \in (\mathbb{C} \cup \{\infty\}) \setminus \Lambda(A)$, where $\Lambda(A)$ is the set of eigenvalues of A, then

$$\mathcal{K}^{\text{rat}}_m(A, v, \sigma) = q_{m-1}(A)^{-1}\mathcal{K}_m(A, v), \qquad q_{m-1}(z) = \prod_{\substack{j=1\\ \sigma_j\neq\infty}}^{m-1} (z - \sigma_j), \tag{1.1}$$

is called a *rational Krylov subspace*. If we set all finite shifts of an $m_\ell + m_r - 1$ dimensional rational Krylov subspace to 0, then the subspace becomes

$$\mathcal{K}_{m_\ell,m_r}(A, v) = \operatorname{span}\left\{A^{-m_r+1}v, \ldots, A^{-1}v, v, Av, A^2v, \ldots, A^{m_\ell-1}v\right\},$$

*The research was partially supported by the Research Council KU Leuven, projects CREA-13-012 Can Unconventional Eigenvalue Algorithms Supersede the State of the Art, OT/11/055 Spectral Properties of Perturbed Normal Matrices and their Applications, CoE EF/05/006 Optimization in Engineering (OPTEC), and fellowship F+/13/020 Exploiting Unconventional QR-Algorithms for Fast and Accurate Computations of Roots of Polynomials; by the DFG research stipend MA 5852/1-1; by the Fund for Scientific Research–Flanders (Belgium) project G034212N Reestablishing Smoothness for Matrix Manifold Optimization via Resolution of Singularities; by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office, Belgian Network DYSCO (Dynamical Systems, Control, and Optimization); and by the Serbian Ministry of Education and Science project #174002 Methods of Numerical and Nonlinear Analysis with Applications.

†Department of Computer Science, KU Leuven, Celestijnenlaan 200A, 3001 Leuven (Heverlee), Belgium; {thomas.mach,raf.vandebril}@cs.kuleuven.be.

‡Department Mathematics and Informatics, University of Banja Luka, M. Stojanovića, 51000 Banja Luka, Bosnia and Herzegovina; pranic77m@yahoo.com.

which is called an *extended Krylov subspace*. Extended Krylov subspaces have been investigated first by Druskin and Knizhnerman in [4]. The advantage over rational Krylov subspace is that only one inverse, factorization, or preconditioner of A is necessary. Extended Krylov subspaces have been proven to be useful, see, e.g., [12, 13, 15], and further allow for symmetric A to use "a short orthogonalization recursion with up to four terms" [4]. On the other hand the additional flexibility of different shifts might be used to achieve the same accuracy with smaller subspaces. Rational Krylov subspaces suffer from the inherent difficulty that one needs good shifts, investigated recently in [10] by Güttel. Rational Krylov subspaces have been used to solve matrix equations, for instance, in the context of model order reduction, see, e.g., [5, 9].

For every Krylov subspace $\mathcal{K}_m(A, v)$ of dimension m a matrix $V \in \mathbb{C}^{n\times m}$ with orthogonal columns exists, so that

$$\operatorname{span}\{V_{:,1:k}\} = \operatorname{span}\left\{v, Av, A^2v, \ldots, A^{k-1}v\right\} \quad \forall k \leq m. \tag{1.2}$$

It is well known that the *projected counterpart* H of A, defined by $H := V^*AV$, is of upper Hessenberg form [8]. Let V now be defined analogously for a rational Krylov subspace with only finite poles, $\mathcal{K}_m^{\text{rat}}(A, v, \sigma)$. Fasino has shown in [6] for A Hermitian that $H = V^*AV$ is of diagonal-plus-semiseparable form, meaning that all submatrices $H_{1:k,k+1:n}$ are of rank 1. If V spans an extended Krylov subspace of the form

$$\operatorname{span}\left\{v, Av, A^{-1}v, A^{-2}v, A^{-3}v, A^2v, A^3v, \ldots\right\},$$

then $H = V^*AV$ is a block upper triangular matrix containing Hessenberg blocks and blocks of inverse Hessenberg form on the block diagonal, see [28]. In Section 2 we will describe the structure of H for rational Krylov subspaces with mixed finite and infinite poles. The matrix $H - D$, where D is a diagonal matrix containing the finite poles, has also only Hessenberg and inverse Hessenberg blocks on the block diagonal [18, Sect. 2.2]. Based on the structure of H we will describe the algorithm and proof an adapted variant of the implicit-Q-theorem.

The main idea of computing approximate, rational Krylov subspaces without inversion is to start with a large Krylov subspace and then perform special similarity transformations that bring the projected matrix H into the desired form. At the end we keep only a small upper left part of H containing the interesting information. For these computations it is not necessary to invert A, to solve with A, nor with $A - \sigma_i I$. We will show that under certain assumptions the computed $\hat{H}$ and $\hat{V}$ approximate the H and V obtained directly from the rational Krylov subspace. This was shown already for extended Krylov subspaces in [17].

Block Krylov subspace methods are an extension, for instance necessary to solve matrix equations with right-hand sides of rank larger than one, see [11, 14]. Instead of using only one vector v, one uses a set of orthogonal vectors $\mathcal{V} = [v_1, v_2, \ldots, v_b]$. The block Krylov subspace then becomes

$$\begin{aligned}\mathcal{K}_m(A, \mathcal{V}) &= \operatorname{span}\left\{\mathcal{V}, A\mathcal{V}, A^2\mathcal{V}, A^3\mathcal{V}, \ldots, A^{m-1}\mathcal{V}\right\} \\ &= \operatorname{span}\left\{v_1, \ldots, v_b, Av_1, \ldots, Av_b, \ldots\right\}.\end{aligned}$$

Block Krylov subspaces can often be chosen of smaller dimension than the sum of the dimension of the Krylov subspaces $\mathcal{K}(A, v_1), \ldots, \mathcal{K}(A, v_b)$, since one uses information from $\mathcal{K}(A, v_i)$ for, e.g., the approximation of $f(A)v_j$. Block extended and block rational Krylov subspaces can be formed by adding negative powers of A such as $A^{-k}\mathcal{V}$ or $\prod_{j=k,\sigma_t\neq\infty}^{1}(A - \sigma_j I)^{-1}\,\mathcal{V}$. Block (rational) Krylov subspace methods have also been used in model order reduction, see, e.g., [1, 3, 7] or more recently for bilinear control systems [2]. We will describe the approximation of block rational Krylov subspaces in Section 3.

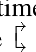
If the matrix A is symmetric or Hermitian[1], then also the matrix $H = V^*AV$ inherits this structure, thus H is tridiagonal. Exploiting the symmetry reduces the computational costs of the algorithm and is discussed in Section 4.

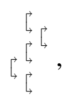
First we introduce the notation and review the essentials about handling rotators.

1.1. Preliminaries. Throughout the paper the following notation is used. We use capital letters for matrices and lower case letters for (column) vectors. Lower case letters are also used as indices. For scalars we use lower case Greek letters. The Hermitian conjugate of a matrix A is denoted by a superscripted asterisk, A^*. Arbitrary entries of matrices are marked with $\times$. To emphasize some of these entries we use $\otimes$. With $I_m \in \mathbb{C}^{m\times m}$ the identity matrix is denoted and with $e_i \in \mathbb{C}^m$ the ith column of I_m. We further use the following calligraphic letters: $\mathcal{O}$ for the big O notation, $\mathcal{K}$ for Krylov subspaces, $\mathcal{V}$ for subspaces, and $\mathcal{E}_k$ for the subspace spanned by the first k columns of the identity matrix.

The following explanations rely heavily on clever manipulations of rotators, therefore we briefly review them. Identity matrices with embedded 2×2 unitary matrices on the diagonal of the form

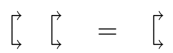
$$\begin{bmatrix} \alpha & \beta \\ -\overline{\beta} & \overline{\alpha} \end{bmatrix},$$

with $|\alpha|^2+|\beta|^2 = 1$ are called *rotators*; sometimes also named *Givens* or *Jacobi rotations* [8]. For the notation of a series of rotators we use  for one rotator. The tiny arrows point to the two rows where the matrix is embedded. If the rotator is applied to a matrix on the right, then the arrows also point to the two rows of the matrix that are changed. Obviously rotators acting on different rows commute. If we have a series of rotators, e.g.,

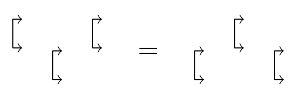


then we call the ordering of the rotators a *shape* or a *pattern* [19].

For handling rotators efficiently we need three operations: the merging, the turnover, and the transfer of rotators through upper triangular matrices. Two rotators acting on the same rows can be *merged*, resulting in a single rotator

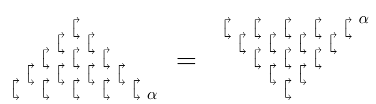


A *turnover* is the operation of three rotators changing their pattern from a V-shaped sequence to an A-shaped sequence (and vice versa)



More generally it is possible to factorize an arbitrary unitary matrix $Q \in \mathbb{C}^{n\times n}$ in $\frac{1}{2}n(n-1)$ rotators and an identity matrix, where one entry is replaced by α, with $\alpha = \det Q$. There are different possible patterns for arranging these rotators, for instance the following two pyramidal shapes [26]:

$$Q = \begin{matrix} \times&\times&\times&\times&\times&\times \\ \times&\times&\times&\times&\times&\times \\ \times&\times&\times&\times&\times&\times \\ \times&\times&\times&\times&\times&\times \\ \times&\times&\times&\times&\times&\times \\ \times&\times&\times&\times&\times&\times \end{matrix} =$$

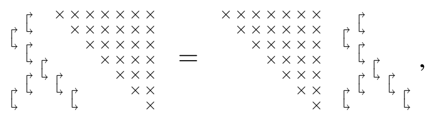


A-pyramidal shape V-pyramidal shape

[1]In the remainder of this paper A symmetric means $A = A^T$ for $A \in \mathbb{R}^{n\times n}$ and $A = A^*$ for $A \in \mathbb{C}^{n\times n}$.

where α stands for the diagonal matrix. The α is not part of the pyramidal shape. If the pyramidal shape points up we call it an A-pyramidal shape, otherwise a V-pyramidal shape. An A-pyramidal shape can always be transformed into a V-pyramidal shape [27, Chapter 9].

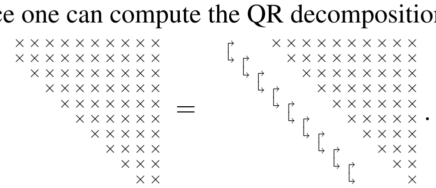
Further, one can transfer rotators through an upper triangular matrix. Therefore one has to apply the rotator to the upper triangular matrix, assume it is acting on row i and $i+1$, this creates an unwanted non-zero entry in position $(i+1, i)$ of the upper triangular matrix. This non-zero entry can be eliminated by applying a rotator from the right, acting on columns i and $i+1$. Transferring rotators one by one, one can pass a whole pattern of rotators through an upper triangular matrix, e.g.,



thereby preserving the pattern of rotations.

In this article we will use the QR decomposition extensively. Moreover, we will factor the unitary Q as a product of rotations. If a matrix exhibits some structure, often also the pattern of rotations in Q's factorization is of a particular shape.

An *upper Hessenberg matrix* H is a matrix where all the entries $H_{i,j}$ with $i > j+1$ are zero. If none of the entries $H_{i+1,i}$ is zero, then H is called an *unreduced Hessenberg matrix*. The QR decomposition of an upper Hessenberg matrix is structured, since with $n-1$ rotators in a descending sequence one can compute the QR decomposition as follows, e.g.,



If the $n-1$ rotators acting on different rows are not ordered in a descending sequence, but in an arbitrary pattern, then we call the matrix an *extended Hessenberg matrix*. If all rotators are non-diagonal we call the extended Hessenberg matrix *unreduced*.

2. Rational Krylov Subspaces. In [17] we have shown how to compute an approximate extended Krylov subspace without explicit inversions. We will now generalize this, starting with the most simplest case–the rational Krylov without blocks or symmetry. The main difference to the algorithm for extended Krylov subspaces is that finite poles can have shifts different from zero. This affects the structure of the matrix $H = V^*AV$ and the algorithm. Further we need an adaption of the implicit-Q-theorem [17, Theorem 3.5], see Theorem 2.2.

2.1. Structure of H. Let $\sigma = [\sigma_1, \sigma_2, \ldots, \sigma_{m-1}]$, with $\sigma_i \in (\mathbb{C} \cup \{\infty\}) \setminus \Lambda(A)$ be the vector of poles. We have two different types of poles, finite and infinite. For the infinite poles we add vectors $A^k v$ to our space and for the finite poles vectors $\left(\prod_{j=k,\sigma_j\neq\infty}^{1}(A-\sigma_j I)^{-1}\right)v$. For $\sigma = [\infty, \sigma_2, \sigma_3, \infty, \ldots]$ the rational Krylov subspace starts with

$$\mathcal{K}_m^{\text{rat}}(A, v, \sigma) = \left\{v, Av, (A-\sigma_2 I)^{-1}v, (A-\sigma_3 I)^{-1}(A-\sigma_2 I)^{-1}v, A^2v, \ldots\right\}. \tag{2.1}$$

The shifts for finite poles provide additional flexibility, which is beneficial in some applications. For the infinite poles we can also shift A and add $(A-\zeta_k)v$ instead, but this provides no additional flexibility, since the spanned space is not changed as the following lemma shows.

LEMMA 2.1. *Let $\mathcal{K}_m(A, v)$ be a standard Krylov subspace of dimension m,*

$$\mathcal{K}_m(A, v) = \operatorname{span}\left\{v, Av, A^2v, \ldots, A^{m-1}v\right\}.$$

The subspaces

$$\operatorname{span}\{\mathcal{K}_m(A,v) \cup \operatorname{span}\{A^m v\}\} \quad \textit{and} \quad \operatorname{span}\{\mathcal{K}_m(A,v) \cup \operatorname{span}\{(A-\zeta_k I)^m v\}\}$$

are identical.

Proof. We have that

$$(A-\zeta_k I)^m v = A^m v + \sum_{j=0}^{m-1} \gamma_j A^j v.$$

Thus the subspaces are identical, since $\sum_{j=0}^{m-1} \gamma_j A^j v \in \mathcal{K}_m(A,v)$. □

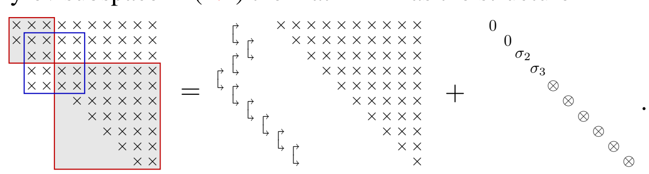
Let V span the rational Krylov subspace as in (1.2) and be $H = V^*AV$. The matrix $H-D$, where D is a diagonal matrix with

$$D_{1,1} = 0 \qquad \text{and} \qquad D_{i,i} = \begin{cases} \sigma_{i-1}, & \sigma_{i-1} \neq \infty, \\ 0, & \sigma_{i-1} = \infty, \end{cases} \qquad i = 2, \ldots, n-1, \tag{2.2}$$

is of extended Hessenberg structure, see [18, Sect. 2.2], [6]. If σ_i is an infinite pole, then the $(i-1)$st rotation is positioned on the left of the ith rotation. Is, however, σ_i finite, then the $(i-1)$st one is on the right of the ith rotation.

For the Krylov subspace in (2.1) the matrix H has the structure

(2.3)



The matrix H consists of overlapping Hessenberg (first and last square) and inverse Hessenberg blocks (second square). For the infinite poles we can add shifts arbitrarily as Lemma 2.1 shows. These shifts are marked with $\otimes$ in the scheme above. For convenience we will choose these poles equal to the last finite one.

2.2. Algorithm. We will now describe how to obtain the structure shown in the example above. The algorithm consists of three steps:

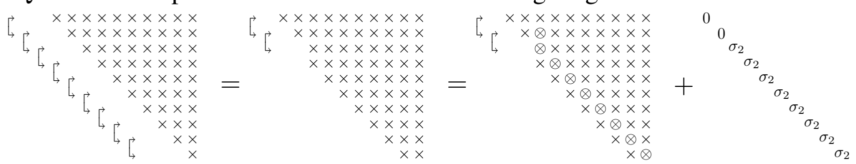
- Construct a large Krylov subspace $\mathcal{K}_{m+p}(A,v)$ spanned by the columns of V and set $H = V^*AV$.
- Transform H into the desired structure.
- Select the upper left $m \times m$ part of H and the first m columns of V.

We will now explain these steps in detail by an example computing the rational Krylov subspace from (2.1). The algorithm starts with a large Krylov subspace $\mathcal{K}_{m+p}(A,v)$. Let the columns of V span $\mathcal{K}_{m+p}(A,v)$ as in (1.2). Then the projection of A on V yields an upper Hessenberg matrix $H = V^*AV$. These matrices fulfill the equation

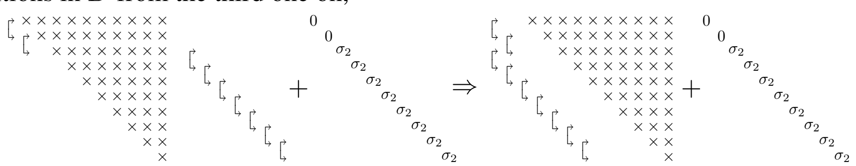
$$AV = VH + r \begin{bmatrix} 0 & 0 & \cdots & 1 \end{bmatrix}, \tag{2.4}$$

where r is the residual. The QR decomposition of H is computed and the Q factor is stored in the form of $n-1$ rotators. In case H is a reduced Hessenberg matrix one has found an associated invariant subspace. An invariant subspace allows to project the original problem. Solving the small dense problem is typically easy and will not be investigated here. Thus we assume that H is unreduced, hence all rotators in Q differ from the identity.

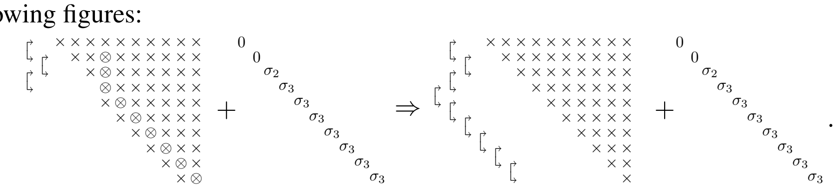
According to Figure (2.3) we keep the first two rotators but have to change the positioning of the third rotator. The third rotator is on the wrong side and thus we have to bring the trailing rotators to the other side. Therefore, we apply all rotators except the first two to R. Because of the descending ordering of the rotators this creates new non-zero entries in the subdiagonal of R. We then introduce the pole σ_2: the diagonal matrix $\mathrm{diag}\,[0, 0, \sigma_2, \ldots, \sigma_2]$ is subtracted from QR. These steps are summarized in the following diagrams:



The elements marked with $\otimes$ are the ones that are changed by introducing the poles. In the next step we restore the upper triangular matrix by applying rotators from the right. These rotations are then brought by a similarity transformation back to the left-hand side. This similarity transformation preserves the structure of D, as the same shift σ_2 appears in all positions in D from the third one on,



The procedure is then repeated for all subsequent poles. The second finite pole provides the following figures:

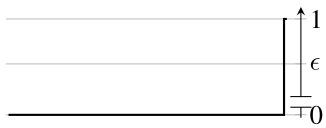


For infinite poles we do not change the pattern. We also do not change the subsequent poles, since this would destroy the matrix structure.

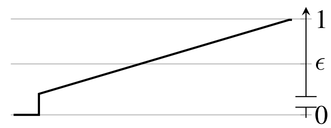
This brings the structure of H in the desired form, but looking on (2.4) we see that the residual also gets affected. The residual R is of rank 1 and has as initial structure

$$R = r \underbrace{\begin{bmatrix} 0 & 0 & \cdots & 1 \end{bmatrix}}_{=:h}.$$

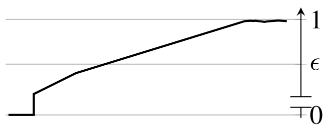
We apply a series of rotators to h and thereby destroy the zero pattern. However, since rotators preserve the norm, we typically observe that the absolute values of the entries in h are decaying from h_n to h_j. (The first entries $h_{1:j-1}$ are not affected by the rotations.) This is sketched in Figure 2.1(a) and (b), where a logarithmic y-axis with an added point for 0 is used. The ϵ marks the accuracy of the used arithmetic, e.g., the IEEE double precision. Every time a finite pole is handled the "energy" of h is distributed more to the left, see Fig. 2.1(c). Finally we retain the first part of V, where the residual is often very small, see Fig. 2.1(d). We choose an oversampling parameter p at the beginning determining how many additional vectors we add to the standard Krylov subspace, since we keep m vectors, we start with $m + p$ ones. By applying the similarity transformations we change V, H, and h in (2.4). At the end we select the leading $m \times m$ block of H. Since for the unreduced H the row $H_{m+1,1:m}$ is not zero, the selection of the first vectors produces a new residual of the form $V_{:,m+1}H_{m+1,1:m}$. Thus, we finally end up with a residual of rank 2. The approximation is successful if the entries



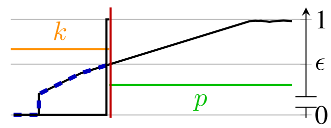
(a) Initial h.



(b) First step.



(c) After 15 steps.



(d) Selecting the first vectors.

Fig. 2.1: Log-scale plots of the residual, showing the effect of the similarity transformation and the selection of the first vectors.

in $h_{1:m}$ (blue dashed part in Fig. 2.1(d)) are (negligible) small. Nevertheless, the algorithm computes only an approximation to the extended Krylov subspaces. The examples and theoretical results in [17], however, show that the approximation for extended Krylov subspaces is often very good. For rational Krylov subspaces this is confirmed by the numerical example in Subsection 2.4.

2.3. Implicit-Q-Theorem. The following variant of the implicit-Q-theorem in [17] shows that the algorithm described in the last subsection leads indeed to an approximation of the sought rational Krylov subspace. It is shown that there is essentially one extended Hessenberg plus diagonal matrix H of the prescribed structure, which is at the same time the projection of A onto V, with $Ve_1 = v$.

THEOREM 2.2. *Let A be a non-singular $n \times n$ matrix, σ and $\hat{\sigma}$ be two shift vectors and let $\underline{V}$ and $\underline{\hat{V}}$ be two $n \times (m+1)$ rectangular matrices having orthonormal columns and sharing the first column $\underline{V}e_1 = \underline{\hat{V}}e_1$. Let V and $\hat{V}$ be the first m columns of $\underline{V}$ and $\underline{\hat{V}}$, respectively. Consider*

$$\begin{aligned} AV &= VH + rw_k^* = \underline{V}\underline{H} = \underline{V}\big(QR + D\big), \\ A\hat{V} &= \hat{V}\hat{H} + \hat{r}\hat{w}_k^* = \underline{\hat{V}}\underline{\hat{H}} = \underline{\hat{V}}\big(\hat{Q}\hat{R} + \hat{D}\big), \end{aligned}$$

where Q and $\hat{Q}$ are decomposed into a series of rotations ordered as imposed by σ and $\hat{\sigma}$. Let further $H - D$ and $\hat{H} - \hat{D}$ be invertible.

Then define $\hat{k}$ as the minimum

$$\hat{k} = \min_i \left\{ 1 \le i \le n-2 \text{ such that, } G_i^Q = I, G_i^{\hat{Q}} = I, \text{ or } \sigma_{i-1} \neq \hat{\sigma}_{i-1} \right\},$$

if no such $\hat{k}$ exists, set it equal to m.

Then the first $\hat{k}$ columns of V and $\hat{V}$, and the upper left $\hat{k} \times \hat{k}$ blocks of V^*AV and $\hat{V}^*A\hat{V}$ are essentially the same, meaning there is a diagonal matrix E, with $|E_{i,i}| = 1$, so that $VE = \hat{V}$ and $E^*V^*AVE = \hat{V}^*A\hat{V}$.

To proof this theorem the following lemma is required, which is the rational Krylov analogue to [28, Thm. 3.7].

LEMMA 2.3. *Let H be an $n \times n$ matrix, with*

$$H = QR + D,$$

where Q is unitary with a decomposition into rotations according to a shift vector σ, R an upper triangular matrix, and D a diagonal matrix containing the poles as in (2.2). Let further $H - D$ be unreduced. Then for $k = 1, \ldots, n-1$,

$$\operatorname{span}\{e_1, \ldots, e_k\} = \mathcal{E}_k = \mathcal{K}_k^{rat}(H, e_1, \sigma),$$

where e_j is the j-th column of the identity matrix.

Proof. First we show analog to [28, Lemma 3.6] that for $k = 1, \ldots, n-2$,
(a) if $\sigma_k = \infty$, then $H\mathcal{K}_k^{\text{rat}}(H, v, \sigma) \subseteq \mathcal{K}_{k+1}^{\text{rat}}(H, v, \sigma)$ and
(b) if $\sigma_k \neq \infty$, then $(H - \sigma_k I)^{-1}\mathcal{K}_k^{\text{rat}}(H, v, \sigma) \subseteq \mathcal{K}_{k+1}^{\text{rat}}(H, v, \sigma)$.
Let

$$\mathcal{K}_k^{\text{rat}}(H, v, \sigma) = \operatorname{span}\left\{ \left(\prod_{j=k, \sigma_j \neq \infty}^{1} (H - \sigma_j I)^{-1} \right) v, \ldots, v, \ldots, H^{q_k} v \right\},$$

with $q_k = |\{i \leq k | \sigma_i = \infty\}|$. Further let u_p be defined for $p \leq k - q_k$ by

$$u_p := \left(\prod_{j=p, \sigma_j \neq \infty}^{1} (H - \sigma_j I)^{-1} \right) v,$$

$p_- := \operatorname{argmax}_{i<p} \sigma_i \neq \infty$, and $p_+ := \operatorname{argmin}_{i>p} \sigma_i \neq \infty$.
If $\sigma_k = \infty$, then $HH^q v = H^{q+1} v$ and $Hu_p = (H - \sigma_{p_-} I)u_p + \sigma_{p_-} u_p \in \operatorname{span}\{u_{p_-}, u_p\}$.
If $\sigma_k \neq \infty$, then

$$\begin{aligned}(H - \sigma_k I)^{-1} H^q v &= (H - \sigma_k I)^{-1}(H - \sigma_k I + \sigma_k I)H^{q-1} \\ &= H^{q-1} v + \sigma_k (H - \sigma_k I)^{-1} H^{q-1} v\end{aligned}$$

and

$$\begin{aligned}(H - \sigma_k I)^{-1} u_p &= (H - \sigma_k I)^{-1}(H - \sigma_{p_+} I)(H - \sigma_{p_+} I)^{-1} u_p. \\ &= u_{p+1} + (\sigma_k - \sigma_{p_+})(H - \sigma_k I)^{-1} u_{p+1}.\end{aligned}$$

Let us now prove the lemma along the argumentation in [28, Thm. 3.7] by induction. The statement is obviously true for $k = 1$. We choose a decompositions of H of the form

$$H = G_L G_k G_R R + D,$$

where G_L and G_R are the rotators to the left and right of G_k respectively, the rotation acting on row k and $k+1$.

Suppose $\sigma_k = \infty$. Using (a) with $v = e_j$, $j \leq k$ provides $H\mathcal{E}_k \subseteq \mathcal{K}_{s,k}^{\text{rat}}(H, e_1, \sigma)$. We will now show that there is an $x \in \mathcal{E}_k$ such that $z = Hx \in \mathcal{E}_{k+1}$ and $e_{k+1}^* z \neq 0$.

We set $x := R^{-1} G_R^{-1} e_k$. Since G_k is not in G_R and R is a non-singular upper triangular $x \in \mathcal{E}_k$. The vector $y := G_k G_R R x$ is in $\mathcal{E}_{k+1}$ and since G_k is non-trivial $e_{k+1}^* y \neq 0$. Further $G_L y \in \mathcal{E}_{k+1}$ since G_{k+1} is not in G_L because of $s_k = \ell$. The vector z defined by

$$z = (G_L G_k G_R R + D)x$$

has the desired structure since D is diagonal with $D_{k+1,k+1} = 0$.

We suppose now $\sigma_k \neq \infty$. Let $y \in \operatorname{span}\{e_k, e_{k+1}\}$ be the solution of $G_k y = e_k$. Since G_k is non-trivial $e_{k+1}^* y \neq 0$. We further have $G_L e_k \in \mathcal{E}_k$ since $s_k = r$. We set $z := R^{-1} G_R^{-1} y \in \mathcal{E}_{k+1}$, with $e_{k+1}^* z \neq 0$ since R^{-1} is invertible. The vector $x := (G_L G_k G_R R + D - \sigma_k I)z$ is in $\mathcal{E}_k$ since $D - \sigma_k I$ is a diagonal matrix with $(D - \sigma_k I)_{k+1,k+1} = 0$. Thus we have a pair (x, z) with $z = (H - \sigma_k I)^{-1} x$. This completes the proof. □

Let us now prove Theorem 2.2.

Proof. The proof is partially analog to [17, Thm. 3.5]. Let $K_n^{\text{rat}}(H, e_1, \sigma)$ be the Krylov matrix having as columns the vectors iteratively constructed for generating the associated Krylov subspace $\mathcal{K}_n^{\text{rat}}(H, e_1, \sigma)$. Then we know from the previous lemma that $K_n^{\text{rat}}(H, e_1, \sigma)$ is upper triangular. Since it holds that

$$\begin{aligned} V K_n^{\text{rat}}(H, e_1, \sigma) = K_n^{\text{rat}}(VHV^*, Ve_1, \sigma) = K_n^{\text{rat}}(A, Ve_1, \sigma) = \\ K_n^{\text{rat}}(A, \hat{V} e_1, \sigma) = K_n^{\text{rat}}(\hat{V}\hat{H}\hat{V}^*, \hat{V} e_1, \sigma) = \hat{V} K_n^{\text{rat}}(\hat{H}, e_1, \sigma) \end{aligned}$$

$V K_n^{\text{rat}}(H, e_1, \sigma)$ and $\hat{V} K_n^{\text{rat}}(\hat{H}, e_1, \sigma)$ are QR decompositions of the same matrix and thus V and $\hat{V}$ and H and $\hat{H}$ are essentially the same for the full dimensional case with identical shift vectors. By multiplication with $P_{\hat{k}} = [e_1, \ldots, e_{\hat{k}}]$ from the right the equality can be restricted to the first $\hat{k}$ columns and the upper left $\hat{k} \times \hat{k}$ block. The cases of $\sigma \neq \hat{\sigma}$ or reduced matrices can thus be handled as in the proof of [17, Thm. 3.5]. □

2.4. Numerical Example. For this and all other numerical experiments in this paper we use MATLAB® implementations of the algorithms. The experiments have been performed on an Intel® Core™i5-3570 (3.40GHz). The following example is an extension of [17, Example 6.5].

EXAMPLE 2.4. We choose $A \in \mathbb{R}^{200 \times 200}$ to be a diagonal matrix with equal distributed eigenvalues $\{0.01, 0.02, \ldots, 2\}$. We used the approximated rational Krylov subspace $\mathcal{K}_k^{\text{rat}}(A, v, \sigma)$ to compute $f(A)v$ by

$$f(A)v \approx V f(H) V^* v = V f(H) e_1 \left\| v \right\|_2,$$

with the columns of $V_{:,1:j}$ spanning $\mathcal{K}_j^{\text{rat}}(A, v, \sigma)$ for all $j \leq k$ and $H = V^* A V$. The entries of the vector v are normal distributed random values with mean 0 and variance 1. To demonstrate the power of shifts we choose a continuous function $f_{[0.10,0.16]}$ focusing on a small part of the spectrum:

$$f_{[0.10,0.16]}(x) = \begin{cases} \exp(-100\,(0.10 - x)), & x < 0.10, \\ 1, & x \in [0.10, 0.16], \\ \exp(-100\,(x - 0.16)), & x > 0.16. \end{cases}$$

For the computations in Figure 2.2 we have chosen the oversampling parameter $p = 100$. The usage of shifts $(0.12, 0.14, 0.16, 0.10)$ improves the accuracy of the computed subspaces significantly. The shifts boost the convergence on the relevant interval $[0.10, 0.16]$. This can also be observed in the plots of the Ritz values in Figure 2.3. In Figure 2.3(a) the Ritz values for the standard Krylov subspace are plotted. Each column in this plot shows the Ritz value of one subspace from dimension 1 to 160. Red crosses stand for Ritz values approximating eigenvalues with an absolute error smaller than $1\,\text{e}{-7.5}$; orange crosses are good approximations with absolute errors between $1\,\text{e}{-7.5}$ and $1\,\text{e}{-5}$; the green crosses are not so good approximations with errors between $1\,\text{e}{-5}$ and $1\,\text{e}{-2.5}$. The typical convergence behavior to the extreme eigenvalues, is observed.

The next Figure 2.3(b) shows the Ritz values of the approximated rational Krylov subspaces computed with our algorithm and the above mentioned shifts. One can clearly see that the shifts enforce the algorithm to store the information for the relevant interval in the first vectors. In and nearby $[0.10, 0.16]$ there are only tiny differences compared with Figure 2.3(c), where we see the Ritz values of the exact rational Krylov subspace.

Finally Figure 2.3(d) shows the Ritz values of the exact extended Krylov subspace. The Ritz values in $[0.10, 0.16]$ approximated the eigenvalues much later than in the previous plot and thus the accuracy of the approximation of $f_{[0.10,0.16]}(A)v$ by an approximated, extended Krylov subspace, red graph in Figure 2.2, is not as good as for the rational Krylov subspace, orange graph.

The first three plots of Figure 2.3 have been merged into a video[2] to make it easier to compare them.

3. The Block Case. If one wants to compute $f(A)v_1, \dots, f(A)v_b$ simultaneously, then one can use a block Krylov subspace of the form

$$\mathcal{K}_m(A, \mathcal{V}) = \operatorname{span}\left\{\mathcal{V}, A\mathcal{V}, A^2\mathcal{V}, A^3\mathcal{V}, \dots, A^{m/b-1}\mathcal{V}\right\}, \quad \text{with} \quad \mathcal{V} = [v_1, \dots, v_b].$$

The dimension of $\mathcal{K}_m(A, \mathcal{V})$ is m and has to be an integer multiple of b.

We will first analyze the structure of the matrix H, the projection of A on the Krylov subspace $\mathcal{K}_k^{\text{rat}}(A, \mathcal{V}, \sigma)$, before we explain the necessary transformations to achieve this structure.

3.1. Structure of H. Let $\mathcal{V}$ be a tall and skinny matrix containing the starting vectors, $\mathcal{V} = [v_1, \dots, v_b] \in \mathbb{C}^{n\times b}$, where b is the block-size[3]. The rational Krylov subspace contains positive powers of A, $A^i\mathcal{V}$, for $\sigma_i = \infty$ and negative powers, $\left(\prod_{t=i,\sigma_t\neq\infty}^{1}(A - \sigma_t I)^{-1}\right)\mathcal{V}$, for $\sigma_i \neq \infty$.

Let $K := K_n^{\text{rat}}(A, \mathcal{V}, \sigma) \in \mathbb{C}^{n\times n}$ be the Krylov matrix of $\mathcal{K}_n^{\text{rat}}(A, \mathcal{V}, \sigma)$. The columns of K are the vectors of $\mathcal{K}_n^{\text{rat}}(A, \mathcal{V}, \sigma)$ without orthogonalization, while the columns of V, defined as in (1.2), form an orthogonal basis of the Krylov subspace. We assume that for all $i \in \{1, \dots, b\}$ the smallest invariant subspace of A containing v_i is $\mathbb{C}^n$. Then there is an invertible, upper triangular matrix U, so that $K = VU$. Since the Krylov subspace is of full dimension, we have $AV = VH$ and $AKU^{-1} = KU^{-1}H$. Setting $H_K := U^{-1}HU$ yields

$$AK = KH_K. \tag{3.1}$$

Since U and U^{-1} are upper triangular the QR decomposition of H has the same zigzag-pattern as H_K. We will derive the structure of H based on the structure of H_K.

3.1.1. The Structure of H_K. We will now describe the structure of H_K and show that the QR decomposition of $H_K - D = Q\tilde{R}$, where D is a diagonal matrix based on the shifts, has a structured zigzag-pattern of rotators. The following example with $\sigma = [\infty, \sigma_2, \sigma_3, \infty, \sigma_5, \infty, \infty, \dots]$ will be used to illustrate the line of arguments. The corresponding Krylov matrix K is then

$$\begin{aligned} K_n^{\text{rat}}(A, \mathcal{V}, \sigma) = \Big[&\mathcal{V}, A\mathcal{V}, (A - \sigma_2 I)^{-1}\mathcal{V}, (A - \sigma_3)^{-1}(A - \sigma_2 I)^{-1}\mathcal{V}, A^2\mathcal{V}, \\ &(A - \sigma_5 I)^{-1}(A - \sigma_3)^{-1}(A - \sigma_2 I)^{-1}\mathcal{V}, A^3\mathcal{V}, A^4\mathcal{V} \dots\Big]. \end{aligned} \tag{3.2}$$

[2] http://people.cs.kuleuven.be/~thomas.mach/ratKrylov/rat_es.mp4

[3] For simplicity we assume that n is an integral multiple of b.

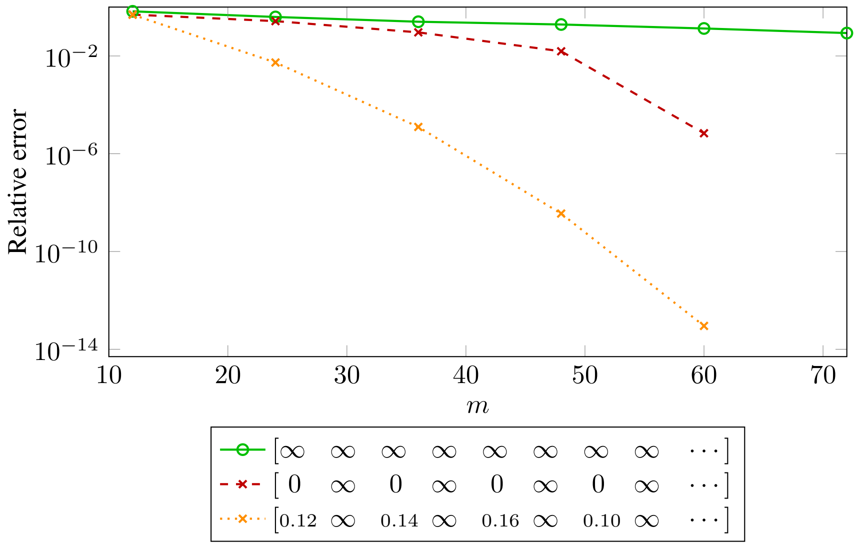


Fig. 2.2: Relative error in approximating $f_{[0.10,0.16]}(A)v$ for $m = 12, 24, 36, 48, 60$.

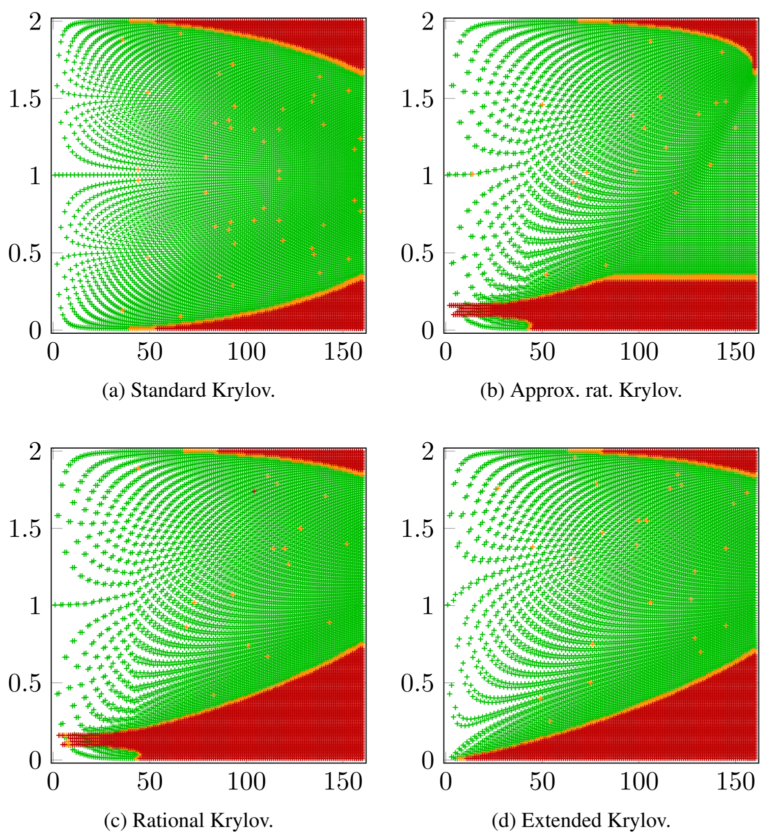


Fig. 2.3: Ritz value plots for equal spaced eigenvalues in $[0, 2]$.

Inserting (3.2) in (3.1) provides

$$
\begin{aligned}
(3.3)\quad K_n^{\text{rat}}(A,\mathcal{V},\sigma)H_K = \Big[& A\mathcal{V}, A^2\mathcal{V}, A(A-\sigma_2 I)^{-1}\mathcal{V}, A(A-\sigma_3)^{-1}(A-\sigma_2 I)^{-1}\mathcal{V}, \\
& A^3\mathcal{V}, A(A-\sigma_5 I)^{-1}(A-\sigma_3)^{-1}(A-\sigma_2 I)^{-1}\mathcal{V}, A^4\mathcal{V}, A^5\mathcal{V}\ldots\Big].
\end{aligned}
$$

The matrix H_K consists of blocks of size $b\times b$. We will now show that H_K in the example is

$$
H_K := \begin{bmatrix}
0 & 0 & I & 0 & 0 & 0 & 0 & \ldots \\
I & 0 & 0 & 0 & 0 & 0 & 0 & \ldots \\
 & 0 & \sigma_2 I & I & 0 & 0 & 0 & \ldots \\
 & 0 & 0 & \sigma_3 I & 0 & I & 0 & \ldots \\
 & I & 0 & 0 & 0 & 0 & 0 & \ldots \\
 & & & & 0 & \sigma_5 I & 0 & \ldots \\
 & & & & I & 0 & 0 & \ldots \\
 & & & & & & I & \ldots
\end{bmatrix}.
$$

One can easily show that for $\sigma_j \neq \infty$

$$
A(A-\sigma_j I)^{-1} \prod_{\substack{t=j-1\\ \sigma_t\neq\infty}}^{1} (A-\sigma_t I)^{-1}\,\mathcal{V} = \sigma_j \prod_{\substack{t=j\\ \sigma_t\neq\infty}}^{1} (A-\sigma_t I)^{-1}\,\mathcal{V} + \prod_{\substack{t=j-1\\ \sigma_t\neq\infty}}^{1} (A-\sigma_t I)^{-1}\,\mathcal{V}
$$

holds. Thus from (3.3) it follows that the diagonal of H_K is D, where D is a diagonal matrix containing the shifts, cf. (2.2),

$$
(3.4)\qquad D = \operatorname{blockdiag}\left(0I_b, \chi_1 I_b, \ldots, \chi_{n-1} I_b\right), \quad \text{with} \quad \chi_i = \begin{cases} \sigma_i, & \sigma_i \neq \infty, \\ 0, & \sigma_i = \infty. \end{cases}.
$$

Let i and j be the indices of two neighbored finite shifts σ_i and σ_j, with $i<j$ and $\sigma_k=\infty$ $\forall i<k<j$, then $H_K(bi+1:b(i+1), bj+1:b(j+1)) = I$. Additionally for j the index of the first finite shift we have $H_K(1:b, bj+1:b(j+1)) = I$.

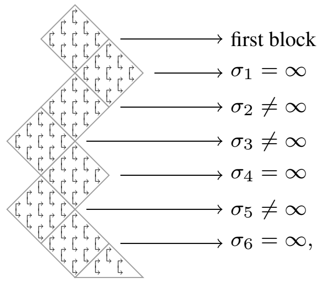
Let q be the index of an infinite shift, then the related columns of K and AK are

$$
K_{:,bq:b(q+1)-1} = A^q\mathcal{V} \qquad \text{and} \qquad AK_{:,bq:b(q+1)-1} = A^{q+1}\mathcal{V}.
$$

Thus for two neighbored infinite shifts $\sigma_i=\infty$ and $\sigma_j=\infty$, with $i<j$ and $\sigma_k\neq\infty$ $\forall i<k<j$, we have $H_K(bj+1:b(j+1), bi+1:b(i+1)) = I$. Additionally for j the index of the first infinite shift we have $H_K(bj+1:b(j+1), 1:b) = I$.

The column of H_K corresponding to the last infinite pole has a special structure related to the characteristic polynomial of A. For simplicity we assume that the last shift is infinite and that the last block column of H_K is arbitrary. We now know H_K.

In the next step we compute the QR decomposition of H_K. Therefore we will assume for an intermediate step that all finite poles are equal to 0. Then the matrix now called H_K^0 in our example has the QR decomposition Q_0R_0. The rhombuses in Q_0 are ordered according to the shift vector σ. For infinite shifts the rhombus is positioned on the right of the previous rhombus and for finite shifts on the left. Thus

$Q_0 =$ 

where all rotators in the rhombuses are $\left[\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right]$ and

$$R_0 = \begin{bmatrix} I & & & \times \\ & \ddots & & \vdots \\ & & I & \times \\ & & & \triangledown \end{bmatrix}.$$

The rotations in the trailing triangle in Q_0 introduce the zeros in the last block column of R_0.

Let us now go back to the rational case with arbitrary finite shifts. Let D be the diagonal matrix defined in (3.4). Then the matrix $H_K - D = H_K^0$ and has thus the same QR decomposition $Q_0 R_0$.

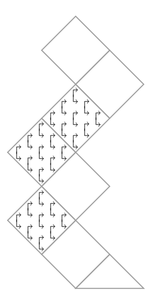
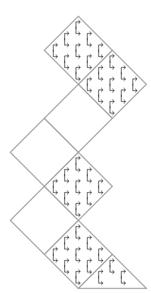
3.1.2. The Structure of H. We will now use the QR decomposition $H_K - D = Q_0 R_0$ to compute the QR decomposition of H. The matrix H is

$$H = U H_K U^{-1} = U\left(Q_0 R_0 U^{-1} + D U^{-1} - U^{-1} D\right) + D,$$

since $D - UU^{-1}D = 0$. The matrix $W = DU^{-1} - U^{-1}D$ is upper triangular. If $\sigma_i = \infty$, then $D_{\rho(i),\rho(i)} = 0$, with $\rho(i)$ the set of indices $\{bi+1, bi+2, \ldots, b\cdot(i+1)\}$ for $i \geq 0$. Thus if $\sigma_i = \infty$ and $\sigma_j = \infty$, then $W_{\rho(i),\rho(j)} = 0$. Further $W_{\rho(i),\rho(i)} = 0$ since $D_{\rho(i),\rho(i)} = \sigma_i I$. In the example W is a block matrix with blocks of size $b \times b$ and the sparsity structure:

$$W = \begin{bmatrix} 0 & 0 & \times & \times & 0 & \times & 0 & 0 \\ & 0 & \times & \times & 0 & \times & 0 & 0 \\ & & 0 & \times & \times & \times & \times & \times \\ & & & 0 & \times & \times & \times & \times \\ & & & & 0 & \times & 0 & 0 \\ & & & & & 0 & \times & \times \\ & & & & & & 0 & 0 \\ & & & & & & & 0 \end{bmatrix}.$$

We now factor $Q_0 = Q_0^r Q_0^\ell$. All blocks, which are on the left of their predecessor, are put into Q_0^r the others into Q_0^ℓ,

$Q_0^r =$  , $Q_0^\ell =$  .

Since Q_0^ℓ consist solely of descending sequences of rhombuses the matrix $H_\ell = Q_0^\ell R_0 U^{-1}$ is of block upper Hessenberg form, in the example:

$$H_\ell = \begin{bmatrix} 0 & 0 & I & & & & & \times \\ I & 0 & 0 & & & & & \times \\ & I & 0 & & & & & \times \\ & & & I & & & & \times \\ & & & & 0 & I & & \times \\ & & & & I & 0 & & \times \\ & & & & & & 0 & \times \\ & & & & & & I & \times \end{bmatrix}.$$

Recall that we can write H as

$$H = U\left(Q_0^r H_\ell + DU^{-1} - U^{-1}D\right) + D = UQ_0^r\left(H_\ell + Q_0^{r*}W\right) + D.$$

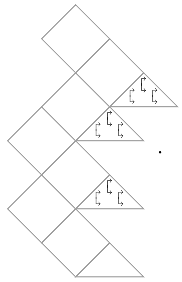
Since W is a block upper triangular with zero block diagonal and Q_0^{r*} contains only descending sequences of rhombuses the product $Q_0^{r*}W$ is block upper triangular, in the example:

$$Q_0^{r*}W = \begin{bmatrix} 0 & 0 & \times & \times & 0 & \times & 0 & 0 \\ & 0 & \times & \times & 0 & \times & 0 & 0 \\ & & 0 & 0 & 0 & \times & 0 & 0 \\ & & & \times & \times & \times & \times & \times \\ & & & & \times & \times & \times & \times \\ & & & & & 0 & 0 & 0 \\ & & & & & & \times & \times \\ & & & & & & & 0 \end{bmatrix}.$$

For $\sigma_i \neq \infty$ we get a non-zero block $(Q_0^{r*}W)_{\rho(i+1),\rho(i+1)}$, since for each $\sigma_i \neq \infty$ the block rows $\rho(i)$ and $\rho(i+1)$ are swapped. However, since $W_{\rho(i+1),\rho(i)} = 0$ the block $(Q_0^{r*}W)_{\rho(i),\rho(i)}$ is zero if additional $\sigma_{i-1} = \infty$. Hence the sum of H_ℓ and $Q_0^{r*}W$ is also block upper Hessenberg with the same block subdiagonal as H_ℓ. In the example the sum is

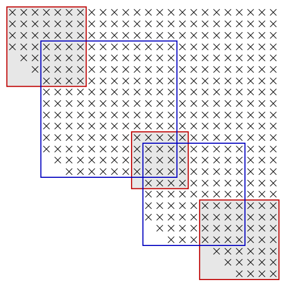
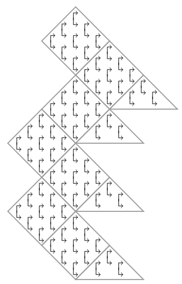
$$H_\ell + Q_0^{r*}W = \begin{bmatrix} 0 & 0 & \otimes & \times & 0 & \times & 0 & \times \\ I & 0 & \times & \times & 0 & \times & 0 & \times \\ & I & 0 & 0 & 0 & \times & 0 & \times \\ & & & \otimes & \times & \times & \times & \times \\ & & & & \times & \otimes & \times & \times \\ & & & & I & 0 & 0 & \times \\ & & & & & & \times & \times \\ & & & & & & I & \times \end{bmatrix}.$$

We now determine $Q_1 = Q_0^r Q_1^\ell Q_1^t$, where Q_1^ℓ and Q_0^ℓ have the same pattern of rotators and Q_1^t will be added later. The rotations in Q_1^ℓ have to be chosen so that $H_\ell + Q_0^{r*}W$ becomes block upper triangular and that the blocks $\rho(i), \rho(i)$ with $\sigma_i = \infty$ or $i = 0$ are also upper triangular. Because of the special structure of $H_\ell + Q_0^{r*}W$ and Q_1^ℓ this is possible. The remaining blocks in the example can be brought to upper triangular form by the rotators in Q_1^t:

$Q_1^t =$  .

After passing Q_1 through the upper triangular matrix U to the right, we have the QR factorization of $H - D$.

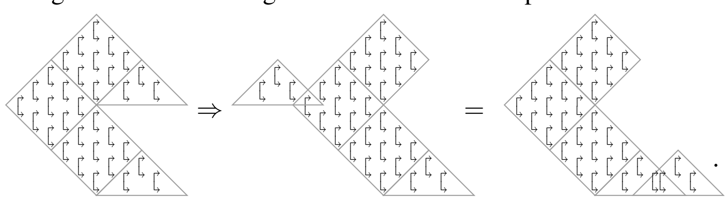
Summarizing the steps above, we have shown that the projection of A on a block rational Krylov subspace like (3.2) spanned by the matrix V leads to a structure like

$H = V^*AV =$  $=$  $R + \operatorname{diag}\begin{smallmatrix} 0\\0\\0\\0\\0\\0\\\sigma_2\\\sigma_2\\\sigma_2\\\sigma_3\\\sigma_3\\\sigma_3\\0\\0\\0\\\sigma_5\\\sigma_5\\\sigma_5\\0\\0\\0\\0\\0\\0 \end{smallmatrix}$,

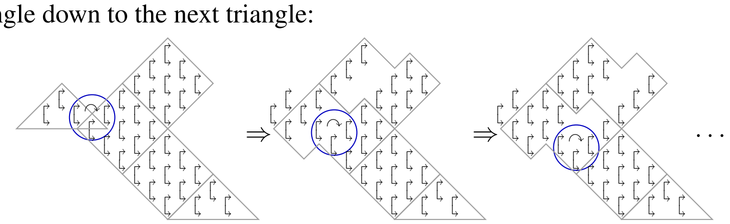
with R an upper triangular matrix.

This structure is not suitable for our algorithm, since the QR decomposition of $H - D$ for the Krylov subspace with solely infinite poles does not have the additional rotators in Q_1^t. We will now show that there are similarity transformations that remove the rotators in Q_1^t. These transformation change the basis of the Krylov subspace but only within the block columns. Thus the approximation properties are not affected if always full blocks are selected.

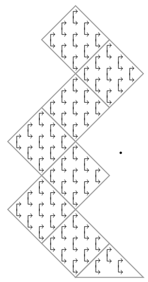
The following three structure diagrams show the main steps:

(3.5) 

First we bring the triangle to the other side. Therefore it has to be passed it through the upper triangular and perform a unitary similarity transformation. This transformation only changes columns within one block. After that, a series of turnovers (blue circles) brings the rotators in the triangle down to the next triangle:

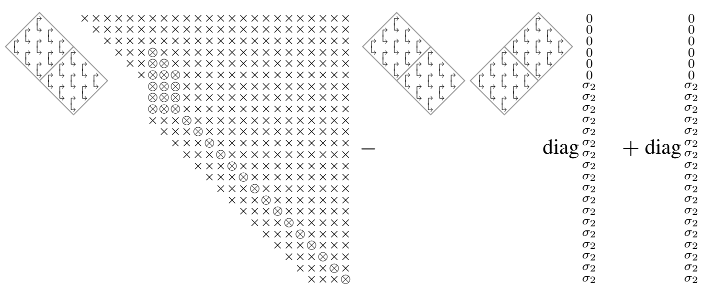
(3.6) 

Finally we can merge the two triangles (in the example for $b = 3$: fuse the rotations in the middle, do a turnover, and fuse the pairs on the left and right). Thus bringing H into a shape without the rotations in Q_1^t is sufficient to approximate the blocks of the block rational Krylov subspace. However we are not able to approximate the individual vectors $\mathcal{K}_n^{\text{rat}}(A,\mathcal{V},\sigma)$ and thus the Krylov condition that $V_{:,1:j}$ spans the first j vectors of $\mathcal{K}_n^{\text{rat}}(A,\mathcal{V},\sigma)$ holds only for $j = ib$, with $i \in \mathbb{N}$. The desired zigzag-shape in our example is:

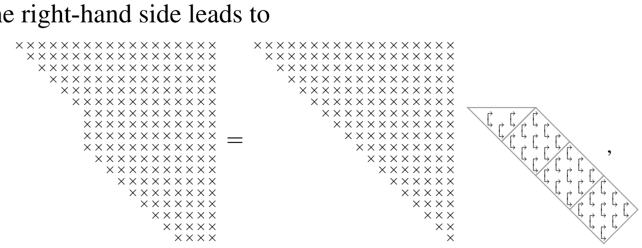


3.2. Algorithm. We can now describe the algorithm to obtain the structure shown in the last subsection. The difference with the algorithm from Subsection 2.2 is that now the rhombuses are arranged according to the shift vector. For each $\sigma_i \neq \infty$ starting with $i = 1$, we have to introduce the pole and to bring all the rhombuses beginning with the $(i+1)$st to the other side. After this has been done for the whole shift vector the first block columns are selected. The approximation is successful if the residual is small enough.

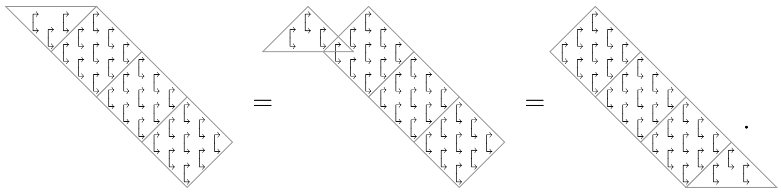
We will now detailed describe how to introduce one pole. If we apply the trailing rotations before introducing the shift, the matrix structure is not perturbed. Since the trailing rhombuses form a descending sequence of rhombuses, applying the rotations to the upper triangular matrix produces an upper Hessenberg matrix with b subdiagonals. Let $\sigma_2 \neq \infty$, and introduce the shift σ_2. The following diagram shows the result of the introduction of the shift:



where the marked entries $\otimes$ represent the non-zero pattern of the second addend. Next the transfer of the rotations is completed by pulling the rotators out to the right. Thereby we restore the upper triangular shape. Unfortunately, this is not as simple as in the one-dimensional case with only one vector. Because of the block structure the zeroing of the entries based on rotators from the right-hand side leads to



where the rotations have the wrong pattern. We have to transform the V-pyramidal shape in the triangle into an A-pyramidal shape and then move the triangle to the lower end by a series of turnovers as in (3.5) and (3.6):



After bringing the rotations on the right-hand side of the upper triangular matrix in the right shape a unitary similarity transformation is used to bring these rotators back to the other side of the matrix. Since this transformation has also to be applied to the diagonal matrix containing the shifts, we have to use the same shift for all trailing positions as in Section 2. Then we continue with the next rhombus. If this rhombus corresponds to an infinite pole, nothing has to be done. Also the shifts in D are not changed, since this would perturb the structure. As we have seen in Lemma 2.1, which also holds for the block case, the shift is not changing the subspace for infinite poles. If this rhombus corresponds to a finite pole, the trailing part of the matrix D is updated to the next shift. The process is continued until the desired shape is retrieved.

3.3. Implicit-Q-Theorem. With the following theorem one can show that in the absence of a residual the above described algorithm computes a block rational Krylov subspace.

THEOREM 3.1. *Let A be a non-singular matrix and σ and $\hat{\sigma}$ be two shift vectors. Let $\underline{V}$ and $\hat{\underline{V}}$ be two $n \times (k+1)b$ rectangular matrices having orthonormal columns sharing the first b columns $\underline{V}[e_1, \ldots, e_b] = \hat{\underline{V}}[e_1, \ldots, e_b] = \mathcal{V}$. Let V and $\hat{V}$ be the first kb columns of $\underline{V}$ and $\hat{\underline{V}}$, respectively. Consider*

$$
\begin{aligned}
AV &= VH + rw_k^* = \underline{V}\,\underline{H} = \underline{V}\big(QR + D\big),\\
A\hat{V} &= \hat{V}\hat{H} + \hat{r}\hat{w}_k^* = \underline{\hat{V}}\,\underline{\hat{H}} = \underline{\hat{V}}\big(\hat{Q}\hat{R} + \hat{D}\big),
\end{aligned}
$$

where Q and $\hat{Q}$ are decomposed into a series of $b \times b$ rhombuses of rotations ordered as imposed by σ and $\hat{\sigma}$. Let further be $H - D$ and $\hat{H} - \hat{D}$ be invertible.

Then define $\hat{k}$ as the minimum index for which one of the $2b^2$ rotations in the ith rhombus of Q or $\hat{Q}$ is the identity or $\sigma_{i-1} \neq \hat{\sigma}_{i-1}$; if no such $\hat{k}$ exists, set it equal to $n-1$.

Then the first $\hat{k}b$ columns of V and $\hat{V}$, and the upper left $\hat{k} \times \hat{k}$ blocks of V^*AV and $\hat{V}^*A\hat{V}$ are block essentially the same. Block essentially the same *means here that $V_{:,jb+1:j(b+1)} = \hat{V}_{:,jb+1:j(b+1)}U$ with $U \in \mathbb{C}^{b\times b}$ and $U^*U = I$.*

The theorem is a generalization of Theorem 2.2. One can prove the theorem analogously to Theorem 2.2 based on an analogue generalization of Lemma 2.3 for the block case. Therefore one has to show first that for $k = 1, \ldots, \frac{n}{b} - 2$,

(a) if $\sigma_k = \infty$, then $H\mathcal{K}_k^{\text{rat}}(H, \mathcal{V}, \sigma) \subseteq \mathcal{K}_{k+1}^{\text{rat}}(H, \mathcal{V}, \sigma)$ and

(b) if $\sigma_k \neq \infty$, then $(H - \sigma_k I)^{-1}\mathcal{K}_k^{\text{rat}}(H, \mathcal{V}, \sigma) \subseteq \mathcal{K}_{k+1}^{\text{rat}}(H, \mathcal{V}, \sigma)$.

The next step is to decompose H into

$$H = G_L G_k G_R R + D,$$

where G_k contains all rotators in the kth rhombus. Based on this decomposition one can proof the block generalization of Lemma 2.3. Finally, the observation that the block QR decomposition is block essential the same (in the above sense) has to be applied to the two block QR decompositions of $K_{s,n}^{\text{rat}}(A, \mathcal{V}, \sigma)$ given in

$$
\begin{aligned}
VK_n^{\text{rat}}(H, [e_1, \ldots, e_b], \sigma) = K_n^{\text{rat}}(VHV^*, V[e_1, \ldots, e_b], \sigma) = K_n^{\text{rat}}(A, \mathcal{V}, \sigma) = \\
K_n^{\text{rat}}(A, \mathcal{V}, \sigma) = K_n^{\text{rat}}(\hat{V}\hat{H}\hat{V}^*, \hat{V}[e_1, \ldots, e_b], \sigma) = \hat{V}K_n^{\text{rat}}(\hat{H}, [e_1, \ldots, e_b], \sigma).
\end{aligned}
$$

Thus ensuring that the computed H has the desired structure is sufficient to compute an approximation to a block rational Krylov subspace. This will be underpinned by the numerical example in the next subsection.

3.4. Numerical Example. The algorithm described above was used to approximately solve a Lyapunov equation

$$AX + XA + BB^* = 0,$$

with the unknown matrix X. The matrix $A \in \mathbb{R}^{5000\times 5000}$ is a diagonal matrix with entries

$$\lambda_i = 5.05 + 4.95\cos(\theta_i), \quad \theta_i \in [0, 2\pi) \quad \forall i,$$

with equal distributed θ_i. The matrix B is of size 5000×2, so that one actually needs a block Krylov algorithm with block-size $b = 2$. The dimension of B is the only point where this example differs from [16, Example 4.2]. The entries of B are computed with the MATLAB command `randn`, meaning they are pseudo-random based on a normal distribution with variance 1 and mean 0. A reference solution is computed with the MATLAB function `lyapchol`, which we assume to be exact. The approximate solution $\tilde{X} \approx X$ is computed based on the projection on an approximated, rational Krylov subspace via

$$\tilde{X} = VYV^*, \text{ where } Y \text{ is the solution of } HY + YH + (V^*B)(V^*B)^* = 0,$$

with $H = V^*AV$. In Figure 3.1 we compare the relative error for B of rank 2 (colored lines, bottom axis) with the results for a B of rank 1 (gray lines, larger marks, top axis). We need for the block-size $b = 2$ about twice as much vectors as for $b = 1$. The oversampling parameter p was chosen to be $100 \cdot b$. To make the comparison easier the gray lines are scaled according to the axis on top.

We observe that the relative accuracy shows almost the same behavior. According to the results from the last section we also observe that the use of shifts (here $\{0.5, 0, 0.25, 0.125\}$ in round robin for the finite poles) improves the accuracy.

4. The Symmetric Case. If the matrix A is symmetric, then the upper Hessenberg matrix $H = V^*AV$ is also symmetric and thus tridiagonal. In this section we will make use of this special structure when computing the approximate extended Krylov subspace. In the symmetric case the same algorithms can be used. Only the representation of the matrix H has to be adapted, so that the symmetry is fully exploited. All results from the previous sections, the implicit-Q-theorem, the structure of H, and the algorithms, stay valid. To exploit the symmetry we replace the QR decomposition of H by the LDL* factorization. Since this is not possible with rotators only, we introduce in the next subsection the eliminators.

4.1. Eliminators. The factorization $H = LDL^*$ of the tridiagonal matrix H is of the form

$$\begin{matrix}\times&\times\\\times&\times&\times\\&\times&\times&\times\\&&\times&\times&\times\\&&&\times&\times&\times\\&&&&\times&\times&\times\\&&&&&\times&\times&\times\\&&&&&&\times&\times&\times\\&&&&&&&\times&\times&\times\\&&&&&&&&\times&\times\end{matrix} = \begin{matrix}1\\\times&1\\&\times&1\\&&\times&1\\&&&\times&1\\&&&&\times&1\\&&&&&\times&1\\&&&&&&\times&1\\&&&&&&&\times&1\\&&&&&&&&\times&1\end{matrix}\begin{matrix}\times\\&\times\\&&\times\\&&&\times\\&&&&\times\\&&&&&\times\\&&&&&&\times\\&&&&&&&\times\\&&&&&&&&\times\\&&&&&&&&&\times\end{matrix}\begin{matrix}1&\times\\&1&\times\\&&1&\times\\&&&1&\times\\&&&&1&\times\\&&&&&1&\times\\&&&&&&1&\times\\&&&&&&&1&\times\\&&&&&&&&1&\times\\&&&&&&&&&1\end{matrix} = \lfloor_{\!\rightarrow}\lfloor_{\!\rightarrow}\lfloor_{\!\rightarrow}\lfloor_{\!\rightarrow}\lfloor_{\!\rightarrow}\lfloor_{\!\rightarrow}\lfloor_{\!\rightarrow}\lfloor_{\!\rightarrow}\lfloor_{\!\rightarrow}\; D \;\lceil^{\!\rightarrow}\lceil^{\!\rightarrow}\lceil^{\!\rightarrow}\lceil^{\!\rightarrow}\lceil^{\!\rightarrow}\lceil^{\!\rightarrow}\lceil^{\!\rightarrow}\lceil^{\!\rightarrow}\lceil^{\!\rightarrow},$$

where we replaced the diagonal matrix in the middle by D to simplify the notation and where $\lfloor_{\!\rightarrow} = \left[\begin{smallmatrix}1&\\\times&1\end{smallmatrix}\right]$ and $\lceil^{\!\rightarrow} = \left[\begin{smallmatrix}1&\times\\&1\end{smallmatrix}\right]$. The brackets here point from the row (no arrow), which is added, to the row (with arrow), which is thus changed. These 2×2-matrices are embedded into large identity matrices. Since these matrices are used in the Gaussian elimination process we call them short *eliminators*. The eliminator $\lfloor_{\!\rightarrow}$ is a *lower eliminator* and $\lceil^{\!\rightarrow}$ an *upper* one.

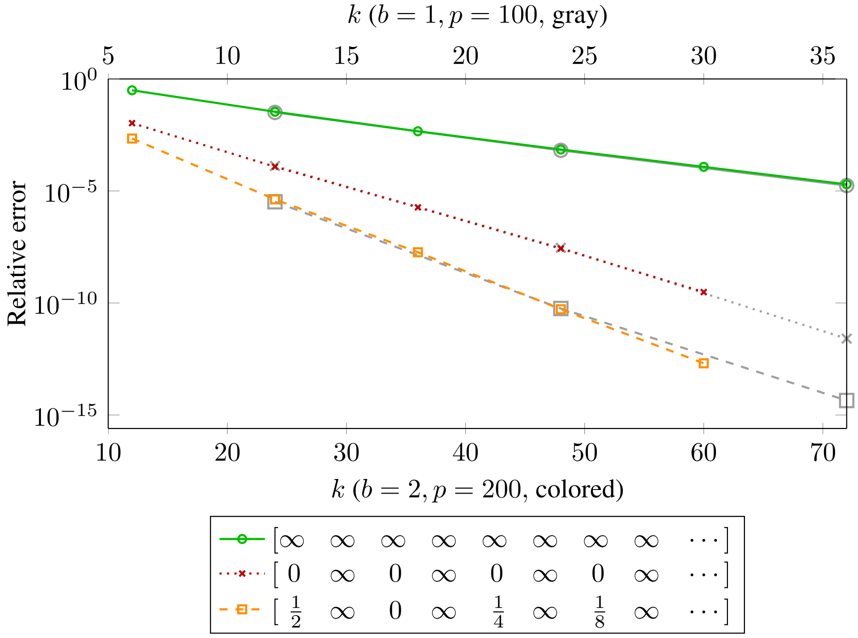


Fig. 3.1: Relative error in approximate solutions of $AX + XA^* + BB^* = 0$ for $k = 12, 24, 36, 48, 60$.

Like with rotators we can perform several operations with eliminators. If the eliminators act on disjoint rows we can change their order arbitrarily. We further can change the ordering in the following cases

$$= \begin{bmatrix} 1 & \xi & \\ & 1 & \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 & & \\ & 1 & \\ & \chi & 1 \end{bmatrix} = \begin{bmatrix} 1 & \xi & \\ & 1 & \\ & \chi & 1 \end{bmatrix} = \quad \text{and} \quad = .$$

We can also pass eliminators through diagonal matrices: applying an eliminator to a diagonal matrix creates a 1×1-bulge; this bulge can be removed by another eliminator acting from the other side. If we want to change the ordering of a lower and an upper eliminator acting on the same rows,

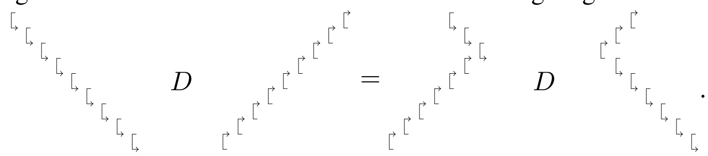
$$\Leftrightarrow ,$$

then we have to compute the product. This gives us a 2×2 matrix. The LDU or UDL factorization provides the desired swap of the eliminators. Looking on the decomposition of the entire matrix we now have an additional diagonal matrix. This small diagonal matrix can be passed through the other eliminators and merged with the main diagonal.

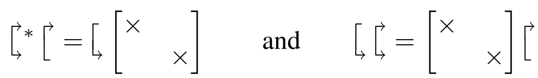
4.2. Algorithm. We will use the same algorithm, but replace the QR decomposition by the LDL* factorization. First, we investigate the non-block, non-rational variant, before we explain the more complex generalizations. We start with a large standard Krylov subspace, with $H = LDL^*$.

The matrix H must now we brought into the shape determined by the shift vector. The lower eliminators on the left have the same zigzag-pattern as the rotations in the general case. The zigzag-pattern for the upper eliminators, however, is flipped.

The transfer of rotators through the upper triangular is replaced by bringing the corresponding lower eliminators from the left to the right and the corresponding upper eliminators from the right to the left. This is summarized in the following diagram for $\sigma = [\infty, 0, \infty, \dots]$,



Now we have to bring the eliminators back to the other side. By a single unitary similarity transformation we can bring both the lower eliminators on the right to the left and the upper eliminators on the left to the right. We want to have



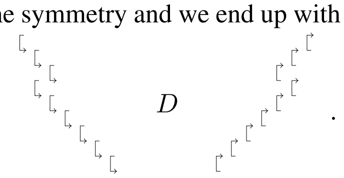
This can be achieved by

$$\begin{bmatrix} \overline{c} & -s \\ \overline{s} & c \end{bmatrix} \begin{bmatrix} 1 & \overline{\xi} \\ & 1 \end{bmatrix} = \begin{bmatrix} 1 & \\ \chi & 1 \end{bmatrix} \begin{bmatrix} d & \\ & e \end{bmatrix}, \qquad \begin{bmatrix} 1 & \\ \xi & 1 \end{bmatrix} \begin{bmatrix} c & s \\ -\overline{s} & \overline{c} \end{bmatrix} = \begin{bmatrix} \overline{d} & \\ & \overline{e} \end{bmatrix} \begin{bmatrix} 1 & \overline{\chi} \\ & 1 \end{bmatrix} \tag{4.1}$$

with

$$c = \frac{1}{\sqrt{1 + |\xi|^2}}, \qquad s = \frac{\overline{\xi}}{\sqrt{1 + |\xi|^2}}, \qquad d = \overline{c}, \qquad e = \overline{s\xi + d}, \quad \text{and} \qquad \chi = \xi.$$

This step also preserves the symmetry and we end up with an LDL* factorization of the form



After we have brought all the eliminators in the right shape determined by the shift vector, we select the first columns and the upper left part of H.

The main advantage of this eliminator based approach is that we make full use of the symmetry. First of all that means that fewer unknown parameters in H, $\mathcal{O}(k+p)$ instead of $\mathcal{O}((k+p)^2)$, are needed. This reduces the number of required floating point operations. For instance, passing one rotator through the upper triangular matrix changes $2(k+p)$ entries in the upper triangular matrix. By using the LDL* factorization we have to change only two entries on the diagonal. The reduced number of floating point operations also reduces the runtime of the algorithm, see Example 4.1. Unfortunately the complexity is not affected, since the most expensive part, the update of V, is in the symmetric case as expensive as in the general case. Second, we preserve the symmetry and can use it in the small computations on the remaining $k \times k$ matrix.

4.3. Numerical Example. The Examples 6.1-4 in [17] are all symmetric. The runtime of the symmetric variant is up to 5% less than the runtime of the non-symmetric implementation used in [17]. This small gain can be explained by the fact that the most expensive step, the update of the subspace V, which is of linear complexity in n, the dimension of A, is the same for the symmetric and the non-symmetric implementation. However, the accuracy of the symmetric variant is almost the same as we will see in the following example.

EXAMPLE 4.1. This example is identical to the 5th example in [13], which has been used as Example 6.3 in [17] in the context of approximate extended Krylov subspaces without explicit inversion.

We compute the product of a matrix function and a vector, $f(A)v$, with $f(x) = 1/\sqrt{x}$, using an approximated, extended Krylov subspace. The matrix A is the discretization of the differential operator $L(u) = \frac{1}{10}u_{xx} - 100u_{yy}$ on the unit square. We use 40 equal spaced interior points. The discretization uses three point stencils in both directions. Together with homogeneous boundary conditions the matrix A is symmetric, positive definite and of size 1600×1600. The vector v is chosen to be $v_j = 1/\sqrt{40}, \forall j$.

We choose the oversampling parameter p to be 200. In Figure 4.1 we can see almost no difference between the symmetric and the non-symmetric implementation; the crosses are always inside the circles. Thus the accuracy of the symmetric variant is as good as the one of the non-symmetric variant.

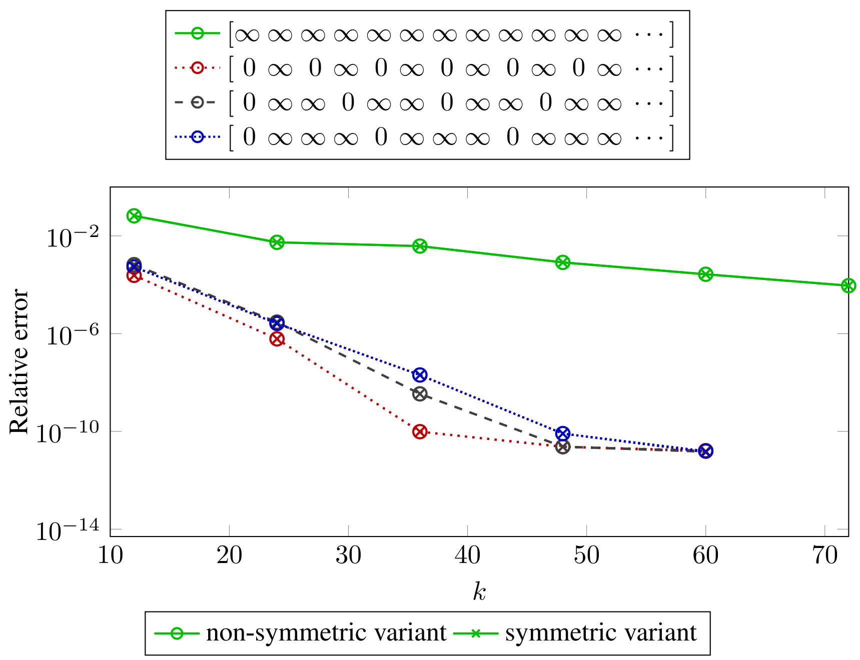
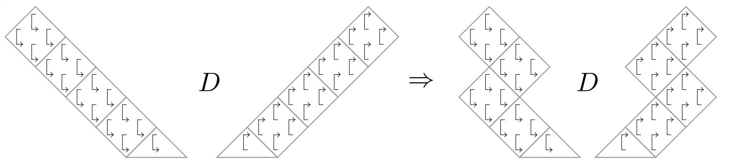


Fig. 4.1: Relative error in approximating $f(x) = 1/\sqrt{x}$ for $k = 12, 24, 36, 48, 60$.

4.4. Combination with Block Rational Krylov Subspaces. Obviously one can combine the ideas for exploiting the symmetry with the algorithm for approximating a block rational Krylov subspace. This leads again only to a different implementation based on the more efficient representation of the symmetric matrix. Thus the theoretical results from Section 2 and 3 remain valid.

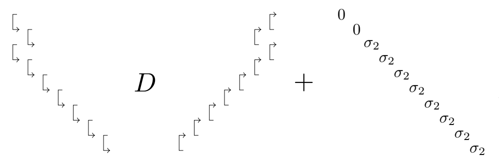
4.4.1. Block case. We will explain the block case for an example with $b = 2$, where we get a pentadiagonal matrix instead of the tridiagonal one as for $b = 1$. Hence the LDL* factorization of this matrix gives us two sequences of eliminators on both sides, which we can group with rhombuses as the rotators in Section 3. Based on the shift vector similarity transformations are used to order the rhombuses on both sides in a way that the result approximates a block extended Krylov subspace. For $\sigma = [\infty, 0, \infty]$ the following diagram sketches the shape:



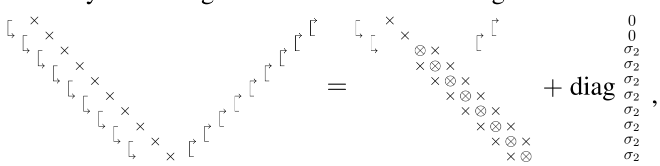
4.4.2. The Rational Case. The LDL* factorization of H of the symmetric rational Krylov subspace

$$\mathcal{K}^{\text{rat}}_{s,k}(A, v, \sigma) = \text{span}\left\{v, Av, (A - \sigma_2 I)^{-1}v, A^2v, A^3v, \ldots\right\} \tag{4.2}$$

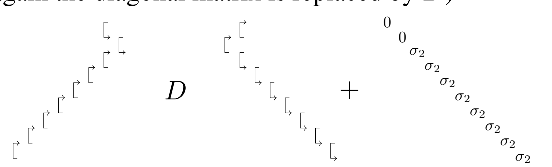
looks like



For the introduction of the shifts a similar trick as for the rational case is used: we apply the trailing eliminators to the diagonal matrix and get a tridiagonal matrix. Then the shifts are introduced and finally the tridiagonal matrix is factorized again. The intermediate step is



where the entries $\otimes$ are changed by introducing the shifts. We observe that the diagonal matrix that is subtracted from the tridiagonal matrix is not changed by applying the inverses of the four eliminators. Next the UDU* factorization of the tridiagonal block is computed. Hence we get (again the diagonal matrix is replaced by D)

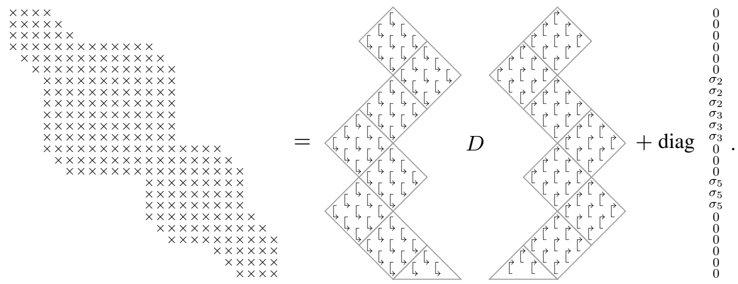


where we now can use rotations to bring the trailing eliminators simultaneously by unitary similarity transformations back to the other side as in (4.1). If the desired rational Krylov subspace has more finite poles the above described steps have to be repeated.

4.4.3. The Symmetric-Block-Rational Case. We just provide an example pattern of a symmetric block rational Krylov subspace for $b = 3$. The necessary steps to achieve this pattern are analogous to the previous sections. The matrix H of the symmetric block rational Krylov subspace

$$\mathcal{K}^{\text{rat}}_{[\ell r r \ell r \ell \ell],8}(A, \mathcal{V}, \sigma) = \text{span}\left\{\mathcal{V}, A\mathcal{V}, (A - \sigma_2 I)^{-1}\mathcal{V}, \ldots\right\}, \tag{4.3}$$

with $A = A^*$ and $\mathcal{V} \in \mathbb{C}^{n\times 3}$ has the factorization:



5. Conclusions. We have presented an algorithm to approximately compute rational Krylov subspaces and extended it to deal with block and/or symmetric Krylov subspaces. The numerical experiments underpin that the algorithm is efficient for some of the examples. However, the algorithm can be interpreted as a compression algorithm operating on an oversampled large Krylov subspace, implying that it cannot add new data in the compression step. Unfortunately this means that the algorithm fails to deliver good results for those applications or examples where the large Krylov subspace lacks the essential information.

Even though this is a major step forward towards an algorithm of practical use, further research is compulsory. Knowing in advance whether the algorithm will work, incorporating preconditioning, and investigating bi-orthogonal Krylov methods are subjected to future investigations.

REFERENCES

[1] A. C. ANTOULAS, *Approximation of Large-Scale Dynamical Systems*, SIAM, Philadelphia, PA, USA, 2005.

[2] T. BREITEN AND T. DAMM, *Krylov subspace methods for model order reduction of bilinear control systems*, Systems and Control Letters, 59 (2010), pp. 443–450.

[3] T. DAMM, *Direct methods and ADI-preconditioned Krylov subspace methods for generalized Lyapunov equations*, Numerical Linear Algebra with Applications, 15 (2008), pp. 853–871.

[4] V. DRUSKIN AND L. KNIZHNERMAN, *Extended Krylov subspaces: Approximation of the matrix square root and related functions*, SIAM Journal on Matrix Analysis and Applications, 19 (1998), pp. 755–771.

[5] V. DRUSKIN AND V. SIMONCINI, *Adaptive rational Krylov subspaces for large-scale dynamical systems*, Systems and Control Letters, 60 (2011), pp. 546–560.

[6] D. FASINO, *Rational Krylov matrices and QR-steps on Hermitian diagonal-plus-semiseparable matrices*, Numerical Linear Algebra with Applications, 12 (2005), pp. 743–754.

[7] R. W. FREUND, *Krylov-subspace methods for reduced-order modeling in circuit simulation*, Journal of Computational and Applied Mathematics, 123 (2000), pp. 395–421.

[8] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, USA, 4th ed., 2013.

[9] S. GUGERCIN, A. C. ANTOULAS, AND C. BEATTIE, *$\mathcal{H}_2$ Model Reduction for large-scale dynamical systems*, SIAM Journal on Matrix Analysis and Applications, 30 (2008), pp. 609–638.

[10] S. GÜTTEL, *Rational Krylov approximation of matrix functions: Numerical methods and optimal pole selection*, GAMM-Mitteilungen, 36 (2013), pp. 8–31.

[11] M. HOCHBRUCK AND G. STARKE, *Preconditioned Krylov subspace methods for Lyapunov matrix equations*, SIAM Journal on Matrix Analysis and Applications, 16 (1995), pp. 156–171.

[12] C. JAGELS AND L. REICHEL, *The extended Krylov subspace method and orthogonal Laurent polynomials*, Linear Algebra and Its Applications, 431 (2009), pp. 441–458.

[13] ———, *Recursion relations for the extended Krylov subspace method*, Linear Algebra and its Applications, 434 (2011), pp. 1716–1732.

[14] K. JBILOU AND A. J. RIQUET, *Projection methods for large Lyapunov matrix equations*, Linear Algebra and its Applications, 415 (2006), pp. 344–358.

[15] L. KNIZHNERMAN AND V. SIMONCINI, *A new investigation of the extended Krylov subspace method for matrix function evaluations*, Numerical Linear Algebra with Applications, 17 (2010), pp. 615–638.

[16] ———, *Convergence analysis of the extended Krylov subspace method for the Lyapunov equation*, Numerische Mathematik, 118 (2011), pp. 567–586.

[17] T. MACH, M. S. PRANIĆ, AND R. VANDEBRIL, *Computing approximate extended Krylov subspaces without explicit inversion*, Department of Computer Science Report TW 623, KU Leuven, 2013.

[18] T. MACH, M. VAN BAREL, AND R. VANDEBRIL, *Inverse eigenvalue problems linked to rational Arnoldi, and rational (non)symmetric Lanczos*, Department of Computer Science Report TW 629, KU Leuven, 2013.

[19] T. MACH AND R. VANDEBRIL, *On deflations in extended QR algorithms*, Department of Computer Science Report TW 634, KU Leuven, 2013.

[20] C. PAIGE AND M. SAUNDERS, *Solution of sparse indefinite systems of linear equations*, SIAM Journal on Numerical Analysis, 12 (1975), pp. 617–629.

[21] A. RUHE, *Rational Krylov sequence methods for eigenvalue computation*, Linear Algebra and its Applications, 58 (1984), pp. 391–405.

[22] ———, *The Rational Krylov algorithm for nonsymmetric eigenvalue problems, III: Complex shifts for real matrices*, BIT, 34 (1994), pp. 165–176.

[23] ———, *Rational Krylov algorithms for nonsymmetric eigenvalue problems, II: Matrix pairs*, Linear Algebra and its Applications, 197/198 (1994), pp. 283–296.

[24] ———, *Rational Krylov: A practical algorithm for large sparse nonsymmetric matrix pencils*, SIAM Journal on Scientific Computing, 19 (1998), pp. 1535–1551.

[25] Y. SAAD, *Krylov subspace methods for solving large unsymmetric linear systems*, Mathematics of Computation, 37 (1981), pp. 105–126.

[26] R. VANDEBRIL, *Chasing bulges or rotations? A metamorphosis of the QR-algorithm*, SIAM Journal on Matrix Analysis and Applications, 32 (2011), pp. 217–247.

[27] R. VANDEBRIL, M. VAN BAREL, AND N. MASTRONARDI, *Matrix Computations and Semiseparable Matrices, Volume I: Linear Systems*, Johns Hopkins University Press, Baltimore, Maryland, USA, 2008.

[28] R. VANDEBRIL AND D. S. WATKINS, *A generalization of the multishift QR algorithm*, SIAM Journal on Matrix Analysis and Applications, 33 (2012), pp. 759–779.